Proyecto Final | Métodos Numéricos para Ingenieros

Introducción

El objetivo de este proyecto final es aplicar y profundizar en el Método de Newton para resolver sistemas de ecuaciones no lineales en el ámbito de las Ciencias de la Computación.

Los estudiantes demostrarán su comprensión tanto teórica como práctica del método mediante la resolución manual de problemas y su implementación computacional. Además, se evaluará la precisión de los resultados obtenidos, utilizando un margen de tolerancia específico.

Objetivos del Proyecto

- **Demostración Teórica:** Realizar manualmente el procedimiento inicial del Método de Newton para tres sistemas de ecuaciones no lineales de dificultad intermedia.
- **Resolución Computacional:** Implementar y resolver los sistemas de ecuaciones no lineales utilizando Excel o un lenguaje de programación (C, Java, Python, etc.).
  - Aplicar el Método de Newton con un margen de tolerancia del 0.001% en el Error Relativo Porcentual para cada iteración.
- **Explicación Paso a Paso:** Elaborar un tutorial detallado, explicando cada paso del proceso de resolución manual y computacional.
- **Reporte Escrito:** Redactar un informe completo con un máximo de 10% de plagio y un 20% de contenido generado por AI, verificado por Turnitin y GPTZero.

Estructura del Proyecto

I. Demostración Teórica

Sistema 1: Optimización en Redes de Transporte

Considere el siguiente sistema de ecuaciones no lineales que modela la optimización en redes de transporte

$$\begin{cases} {x_1}^2 + {x_2}^2 + x_3 - 4 = 0 \\ x_1 sen(x_2) - {x_3}^2 = 1 \\ \log(x_1 x_2) + x_3 - 1 = 0 \end{cases}$$

Procedimiento Manual:

1. Definición de la Función Vectorial
2. Cálculo de la Matriz Jacobiana
3. Iteración del Método de Newton

Procedimiento Computacional

- Aproximación inicial: $x_0 = (1,1,1)^T$.
- Evaluación de $F(x_0)$ y $J(x_0)$.
- Resolución del sistema lineal: $J(x_0)\Delta x = -F(x_0)$.
- Actualización de la aproximación: $x_1 = x_0 + \Delta x$.
- Repetición hasta obtener un margen de tolerancia del 0.001%

Sistema 2: Modelado de Flujo de Fluidos

Considere el siguiente sistema de ecuaciones no lineales que modela el flujo de fluidos en una tubería:

$$\begin{cases} {x_1}^3 - x_2 \cos(x_3) = 2 \\ {x_1}^2 + {x_2}^2 + {x_3}^2 = 3 \\ e^{x_1} + x_2 - x_3 = 1 \end{cases}$$

Procedimiento Manual:

1. Definición de la Función Vectorial
2. Cálculo de la Matriz Jacobiana
3. Iteración del Método de Newton

Procedimiento Computacional

1. Aproximación inicial: $x_0 = (1,1,1)^T$.
2. Evaluación de $F(x_0)$ y $J(x_0)$.
3. Resolución del sistema lineal: $J(x_0)\Delta x = -F(x_0)$.
4. Actualización de la aproximación: $x_1 = x_0 + \Delta x$.
5. Repetición hasta obtener un margen de tolerancia del 0.001%

Sistema 3: Dinámica de Ecosistemas

Considere el siguiente sistema de ecuaciones no lineales que modela la dinámica de un ecosistema con tres especies interdependientes:

$$\begin{cases} x_1 \log(x_2) + x_2 x_3 = 3 \\ {x_1}^2 + sen(x_2) - x_3 = 1 \\ e^{x_1 x_2} + {x_3}^2 = 4 \end{cases}$$

Procedimiento Manual:

1. Definición de la Función Vectorial
2. Cálculo de la Matriz Jacobiana
3. Iteración del Método de Newton

Procedimiento Computacional

2. Aproximación inicial: $x_0 = (1,1,1)^T$.
3. Evaluación de $F(x_0)$ y $J(x_0)$.
4. Resolución del sistema lineal: $J(x_0)\Delta x = -F(x_0)$.
5. Actualización de la aproximación: $x_1 = x_0 + \Delta x$.

6. Repetición hasta obtener un margen de tolerancia del 0.001%

III. Tutorial Paso a Paso

Para cada sistema, incluya una explicación detallada de cada paso del proceso de resolución manual y computacional, incluyendo:

1. **Definición de la función vectorial:** $F(x)$ y cálculo de la matriz jacobiana $J(x)$.
2. **Selección de una aproximación inicial:** $\boldsymbol{x_0}$.
3. **Evaluación de $\boldsymbol{F}(\boldsymbol{x})$ y $\boldsymbol{J}(\boldsymbol{x})$ en cada iteración**.
4. **Resolución del sistema lineal**: $J(x_n)\Delta x_n = -F(x_n)$.
5. **Actualización de la aproximación:** $x_{n+1} = x_n + \Delta x_n$.
6. **Criterio de convergencia**: Verificar si $\| \Delta x_n \| < tolerancia$ o $\| F(x_n) \|$ es suficientemente pequeño.
7. **Explicación de los resultados obtenidos** y comparación con los resultados teóricos esperados.

IV. Error Relativo Porcentual

En cada iteración, calcule el Error Relativo Porcentual (ERP) para cada variable:

$$ERP = \left| \frac{\Delta x}{x} \right| \times 100\%$$

Asegúrese de que el ERP esté por debajo del 0.001% para considerar que el método ha convergido.

V. Reporte Escrito

El reporte final debe incluir:

1. **Introducción:** Descripción del proyecto y objetivos.
2. **Desarrollo Teórico:** Procedimientos manuales detallados para los tres sistemas.
3. **Resolución Computacional:** Implementación en Excel o código fuente en el lenguaje de programación elegido.

4. **Análisis de Resultados:** Comparación de resultados manuales y computacionales, discusión sobre la convergencia y precisión.
5. **Conclusiones:** Reflexiones finales y posibles extensiones del trabajo.
6. **Referencias Bibliográficas:** Citar todas las fuentes utilizadas en el proyecto.

Referencias Bibliográficas

1. Burden, R. L., & Faires, J. D. (2011). **Numerical Analysis**. Brooks/Cole.
2. Chapra, S. C., & Canale, R. P. (2014). **Numerical Methods for Engineers**. McGraw-Hill.
3. Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (2007). **Numerical Recipes: The Art of Scientific Computing**. Cambridge University Press.
4. Ralston, A., & Rabinowitz, P. (2001). **A First Course in Numerical Analysis**. Dover Publications.

Lecture Notes | Método de Newton

Importancia de los Sistemas de Ecuaciones No Lineales

Los sistemas de ecuaciones no lineales son fundamentales en numerosas disciplinas debido a su capacidad para modelar fenómenos complejos que no pueden ser descritos adecuadamente por sistemas lineales. Algunos ejemplos incluyen:

- **Física:** Modelos de dinámica de fluidos, mecánica cuántica y sistemas caóticos.
- **Ingeniería:** Diseño de circuitos, análisis estructural y control de procesos.
- **Biología:** Modelado de crecimiento poblacional, dinámica de enfermedades y procesos bioquímicos.
- **Economía:** Modelos de equilibrio de mercado, teoría de juegos y optimización de recursos.

En Ciencias de la Computación, los sistemas de ecuaciones no lineales aparecen en diversas áreas, como la optimización de redes neuronales, la eficiencia de algoritmos de búsqueda y el modelado de redes de comunicaciones.

Fundamentos Teóricos

Definición de Sistemas de Ecuaciones No Lineales

Un sistema de ecuaciones no lineales consiste en un conjunto de ecuaciones en las que al menos una de las ecuaciones es no lineal. Formalmente, un sistema de ecuaciones no lineales puede escribirse como:

$$\begin{cases} f_1(x_1, x_2, \dots., x_n) = 0 \\ f_2(x_1, x_2, \dots., x_n) = 0 \\ \vdots \\ f_m(x_1, x_2, \dots., x_n) = 0 \end{cases}$$

Donde f_i son funciones no lineales de las variables $x_1, x_2, \dots., x_n$.

Ejemplo de un Sistema No Lineal

Consideremos el siguiente sistema de dos ecuaciones no lineales con dos variables:

$$\begin{cases} x^2 + y^2 - 4 = 0 \\ e^x - y = 0 \end{cases}$$

Este sistema tiene dos ecuaciones: una cuadrática y una exponencial, lo que lo hace no lineal.

Comparación con Sistemas Lineales

En contraste, un sistema de ecuaciones lineales se escribe como:

$$Ax = b$$

Donde:

A es una matriz de coeficientes

x es el vector de variables

b es el vector de términos independientes.

Las ecuaciones lineales son más sencillas de resolver porque no presentan las complicaciones de la no linealidad.

Método de Newton para una Sola Ecuación

Para una sola ecuación no lineal $f(x) = 0$, el Método de Newton (también conocido como Método de Newton-Raphson) es una técnica iterativa que aproxima la solución. La fórmula iterativa para actualizar la aproximación x_n es:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Donde $f'(x_n)$ es la derivada de $f(x_n)$.

Extensión del Método a Sistemas de Ecuaciones

Para extender el Método de Newton a sistemas de ecuaciones no lineales, consideramos un sistema $F(x) = 0$, donde:

$$F(x) = \begin{cases} f_1(x_1, x_2, \dots, x_n) = 0 \\ f_2(x_1, x_2, \dots, x_n) = 0 \\ \vdots \\ f_m(x_1, x_2, \dots, x_n) = 0 \end{cases}$$

Matriz Jacobiana

La matriz jacobiana $J(x)$ es una matriz de derivadas parciales de las funciones f_i con respecto a las variables x_j:

$$J(x) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_n} \end{pmatrix}$$

Fórmula Iterativa del Método de Newton

La fórmula iterativa para actualizar la aproximación x_n es:

$$x_{n+1} = x_n - J(x)^{-1} F(x_n)$$

Descripción Paso a Paso

1. **Calcular $F(x_n)$:**
   - Evaluar las funciones fi en la aproximación actual x_n .
2. **Calcular $J(x)$:**
   - Evaluar la matriz jacobiana en la aproximación actual x_n.
3. **Resolver el Sistema Lineal:**
   - Resolver el sistema $J(x_n)\Delta x = -F(x_n)$ para obtener Δx.
4. **Actualizar la Aproximación:**
   - Actualizar la aproximación con $x_{n+1} = x_n + \Delta x$.

5. **Repetir:**
   - Repetir los pasos anteriores hasta que $\parallel \Delta x \parallel$ sea menor que un umbral de tolerancia o $\parallel F(x_n) \parallel$ sea suficientemente pequeño.

Ejemplo Práctico

Supongamos el siguiente sistema de ecuaciones no lineales:

$$\begin{cases} x^2 + y^2 - 4 = 0 \\ e^x - y = 0 \end{cases}$$

1. **Definición de las Funciones:**

$$F(x) = \begin{pmatrix} x^2 & +y^2 & -4 \\ e^x & -y & \end{pmatrix}$$

2. **Cálculo de la Matriz Jacobiana:**

$$J(x) = \begin{pmatrix} 2x & 2y \\ e^x & -1 \end{pmatrix}$$

3. **Iteración del Método de Newton:**

Supongamos una aproximación inicial $x_0 = (1,1)^T$.

- **Paso 1: Calcular $\boldsymbol{F(x_0)}$:**

$$F(x_0) = \begin{pmatrix} (1)^2 & (1)^2 \\ (e^1) & -1 \end{pmatrix} = \begin{pmatrix} & -2 \\ e & -1 \end{pmatrix}$$

- **Paso 2: Calcular $\boldsymbol{J(x_0)}$:**

$$J(x_0) = \begin{pmatrix} (2)(1) & (2)(1) \\ (e^1) & -1 \end{pmatrix} = \begin{pmatrix} 2 & 2 \\ e & -1 \end{pmatrix}$$

- **Paso 3: Resolver el Sistema Lineal:**

$$J(x_0)\Delta x = -F(x_0)$$

$$\begin{pmatrix} 2 & 2 \\ e & -1 \end{pmatrix} = \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} & 2 \\ 1 & -e \end{pmatrix}$$

Resolviendo, obtenemos:

$$\Delta x \approx \begin{pmatrix} 0.07039 \\ -1.03519 \end{pmatrix}$$

- **Paso 4: Actualizar la Aproximación:**

$$x_1 = x_0 + \Delta x \approx \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 0.07039 \\ -1.03519 \end{pmatrix} = \begin{pmatrix} 1.07039 \\ -0.03519 \end{pmatrix}$$

- **Paso 5: Repetir:**

Repetir los pasos anteriores hasta que se cumpla el criterio de convergencia.

Estructura General del Algoritmo

El Método de Newton para sistemas de ecuaciones no lineales sigue los siguientes pasos:

1. **Inicialización**: Elegir una aproximación inicial x_0.
2. **Iteración**:
    1. Evaluar $F(x_n)$.
    2. Evaluar $J(x_n)$.
    3. Resolver el sistema lineal $J(x_n)\Delta x = -F(x_n)$ para obtener Δx.
    4. Actualizar la aproximación: $x_{n+1} = x_n + \Delta x$.
3. **Criterio de Convergencia**: Repetir el paso 2 hasta que $\| \Delta x \|$ sea menor que un umbral de tolerancia ϵ o $\| F(x_n) \|$ sea suficientemente pequeño.

Ejemplo Práctico

Estos ejemplos están orientados al área de Ciencias de la Computación y cubren diversos contextos prácticos. Para cada ejemplo, se presentará la formulación del problema y se desarrollará el planteamiento general del Método de Newton paso a paso.

Ejemplo 1: Optimización de Redes Neuronales

Consideramos un sistema de ecuaciones no lineales que representa la interacción de pesos en una red neuronal simple:

$$\begin{cases} x_1 \tanh(x_2 x_3) - x_4 = 0 \\ x_2 sen(x_3) - x_1 x_4 + 1 = 0 \\ e^{x_1 x_2} + {x^2}_3 - x_4 = 0 \\ {x^2}_1 + {x^2}_2 + {x^2}_3 + {x^2}_4 - 1 = 0 \end{cases}$$

Paso 1: Definición de la Función Vectorial $\boldsymbol{F(x)}$

La función vectorial $F(x)$ es:

$$F(x) = \begin{pmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \\ f_4(x) \end{pmatrix} = \begin{pmatrix} x_1 \tanh(x_2 x_3) - x_4 \\ x_2 sen(x_3) - x_1 x_4 + 1 \\ e^{x_1 x_2} + {x^2}_3 - x_4 \\ {x^2}_1 + {x^2}_2 + {x^2}_3 + {x^2}_4 - 1 \end{pmatrix}$$

Paso 2: Cálculo de la Matriz Jacobiana $\boldsymbol{J(x)}$

La matriz jacobiana $J(x)$ se obtiene derivando cada componente de $F(x)$ con respecto a cada variable x_i:

$$J(x) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \frac{\partial f_1}{\partial x_3} & \frac{\partial f_1}{\partial x_4} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_2}{\partial x_3} & \frac{\partial f_2}{\partial x_4} \\ \frac{\partial f_3}{\partial x_1} & \frac{\partial f_3}{\partial x_2} & \frac{\partial f_3}{\partial x_3} & \frac{\partial f_3}{\partial x_4} \\ \frac{\partial f_4}{\partial x_1} & \frac{\partial f_4}{\partial x_1} & \frac{\partial f_4}{\partial x_3} & \frac{\partial f_4}{\partial x_4} \end{pmatrix}$$

Calculamos cada derivada parcial:

$$\frac{\partial f_1}{\partial x_1} = \tanh(x_2 x_3)$$

$$\frac{\partial f_1}{\partial x_2} = x_1(1 - tanh^2(x_2 x_3))x_3$$

$$\frac{\partial f_1}{\partial x_3} = x_1 1 - tanh^2(x_2 x_3))x_2$$

$$\frac{\partial f_1}{\partial x_4} = -1$$

$$\frac{\partial f_2}{\partial x_1} = -x_4$$

$$\frac{\partial f_2}{\partial x_2} = sen(x_3)$$

$$\frac{\partial f_2}{\partial x_3} = x_2 \cos{(x_3)}$$

$$\frac{\partial f_2}{\partial x_4} = -x_1$$

$$\frac{\partial f_3}{\partial x_1} = x_2 e^{x_1 x_2}$$

$$\frac{\partial f_3}{\partial x_2} = x_1 e^{x_1 x_2}$$

$$\frac{\partial f_3}{\partial x_3} = 2x_3$$

$$\frac{\partial f_3}{\partial x_4} = -1$$

$$\frac{\partial f_4}{\partial x_1} = 2x_1$$

$$\frac{\partial f_4}{\partial x_2} = 2x_2$$

$$\frac{\partial f_4}{\partial x_3} = 2x_3$$

$$\frac{\partial f_4}{\partial x_4} = 2x_4$$

Entonces, la matriz jacobiana es:

$$J(x) = \begin{pmatrix} \tanh(x_2 x_3) & x_1(1 - tanh^2(x_2 x_3))x_3 & x_1 1 - tanh^2(x_2 x_3))x_2 & -1 \\ -x_4 & sen(x_3) & x_2 \cos(x_3) & -x_1 \\ x_2 e^{x_1 x_2} & x_1 e^{x_1 x_2} & 2x_3 & -1 \\ 2x_1 & 2x_2 & 2x_3 & 2x_4 \end{pmatrix}$$

Paso 3: Iteración del Método de Newton

Supongamos una aproximación inicial $x_0 = (0.5 \quad 0.5 \quad 0.5 \quad 0.5)^T$

1. **Evaluar $F(x_0)$:**

$$F(x_0) = \begin{pmatrix} (0.5)\tanh\big((0.5)(0.5)\big) - 0.5 \\ (0.5)sen(0.5) - (0.5)(0.5) + 1 \\ e^{(0.5)(0.5)} + (0.5)^2 - 0.5 \\ (0.5)^2 + (0.5)^2 + (0.5)^2 + (0.5)^2 - 1 \end{pmatrix}$$

Evaluando los valores, obtenemos:

$$F(x_0) = \begin{pmatrix} -0.37755 \\ 0.9897 \\ 1.0340 \\ 0 \end{pmatrix}$$

2. **Evaluar $\boldsymbol{J(x_0)}$:**

$$J(x_0) = \begin{pmatrix} \tanh(0.25) & (0.5)(1-tanh^2(0.25))(0.5) & (0.5)(1-tanh^2(0.25))(0.5) & -1 \\ -0.5 & sen(0.5) & (0.5)\cos(0.5) & -0.5 \\ (0.5)e^{0.25} & (0.5)e^{0.25} & (2)(0.5) & -1 \\ (2)(0.5) & (2)(0.5) & (2)(0.5) & (2)(0.5) \end{pmatrix}$$

Evaluando los valores, obtenemos:

$$J(x_0) \approx \begin{pmatrix} 0.2449 & 0.2358 & 0.2358 & -1 \\ -0.5 & 0.4794 & 0.4388 & -0.5 \\ 0.6420 & 0.6420 & 1 & -1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

3. **Resolver el Sistema Lineal**

$$J(x_0)\Delta x = -F(x_0)$$

$$\begin{pmatrix} 0.2449 & 0.2358 & 0.2358 & -1 \\ -0.5 & 0.4794 & 0.4388 & -0.5 \\ 0.6420 & 0.6420 & 1 & -1 \\ 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} \Delta x_1 \\ \Delta x_2 \\ \Delta x_3 \\ \Delta x_4 \end{pmatrix} = \begin{pmatrix} -0.37755 \\ 0.9897 \\ 1.0340 \\ 0 \end{pmatrix}$$

4. **Actualizar la aproximación:**

$$x_1 = x_0 + \Delta x$$

5. **Repetir**:

Repetir los pasos anteriores hasta que se cumpla el criterio de convergencia.